

I was finally able to get some better network performance using the Intel Pro/10GbE 10GigE NIC. Between two dual 3.06 GHz Xeon PCs, by upping the PCI-X burst size from 512 to 4096 with:

```
setpci -d 8086:1048 e6.b=2e
```

and modifying the ixgb module parameters with (the most important is the RxIntDelay=0):

```
modprobe ixgb RxIntDelay=0 TxDescriptors=2048 RxDescriptors=2048  
XsumTX=1 XsumRX=1 FlowControl=3
```

I was able to finally break the 5 Gbps barrier on a single stream:

```
[root@chance2 root]# /usr/local/bin/nuttcp -w2048 192.168.10.75  
5986.1875 MB / 10.00 sec = 5021.7016 Mbps 99 %TX 70 %RX
```

This makes me feel much better as my goal all along was to reach at least the 50% utilization level with these first generation 10GigE NICs.

The bidirectional test does slightly better achieving an aggregate of 5.7 Gbps (remember that 133-MHz/64-bit PCI-X has a theoretical upper limit of 8.5 Gbps):

```
[root@chance2 root]# /usr/local/bin/nuttcp -w2048 192.168.10.75 & /usr/local/bin/nuttcp  
-r -w2048 192.168.10.75  
3235.8750 MB / 10.00 sec = 2714.6580 Mbps 99 %TX 66 %RX  
3623.0625 MB / 10.00 sec = 3039.1469 Mbps 84 %TX 88 %RX
```

The above tests were of course with a 9000 byte MTU. For grins I also did a test using an MTU of 16114 (PCs directly connected back to back).

The single stream achieves 5.3 Gbps:

```
[root@chance2 root]# /usr/local/bin/nuttcp -w2048 -T20 192.168.10.75  
12722.6875 MB / 20.00 sec = 5335.4621 Mbps 95 %TX 52 %RX
```

And the bidirectional test manages to crack the 6 Gbps barrier:

```
root@chance2 root]# /usr/local/bin/nuttcp -w2048 -T20 192.168.10.75 &  
/usr/local/bin/nuttcp -r -T20 -w2048 192.168.10.75  
7265.2500 MB / 20.00 sec = 3046.9198 Mbps 78 %TX 80 %RX  
7087.0625 MB / 20.00 sec = 2972.0617 Mbps 86 %TX 58 %RX
```

I'm still at a loss as to why the dual 2 GHz G5s are slower performers.

Using the same procedures as for the PCs, the network throughput is slightly less than 4 Gbps:

```
[root@zeus ~]# /usr/local/bin/nuttcp -w2048 192.168.10.39  
4699.5625 MB / 10.01 sec = 3939.8447 Mbps 99 %TX 78 %RX
```

This is a puzzle as the G5 CPU is faster, it has faster memory access, and the system bus is faster, so I don't understand what's holding it back. Since the CPU is maxed out, perhaps the interrupt overhead on PPC is higher than on the PC.

I wanted to try out the new 2.6 kernel network API (NAPI) on the G5s to see if that would help, but when I enabled that option for the ixgb driver, the ifconfig up on the 10GigE interface would just hang.

Oh well, I was just excited to finally break the 5 Gbps barrier, and wanted to share the good news!

-Bill